SVCBench: A Streaming Video Counting Benchmark for Spatial-Temporal State Maintenance

Pengyiang Liu[1,2], Zhongyue Shi[1], Hongye Hao[1], Qi Fu[1], Xueting Bi[1], Siwei Zhang[1], Xiaoyang Hu[1], Zitian Wang[3†], Linjiang Huang[1], and Si Liu[1]

[1] Institute of Artificial Intelligence, Beihang University, Beijing, China
[2] Meituan, Beijing, China
[3] Hangzhou International Innovation Institute of Beihang University, Hangzhou, China
{buaaplay, wangzt.kghl}@gmail.com, {ljhuang, liusi}@buaa.edu.cn

Project Page: https://buaa-colalab.github.io/SVCBench

Abstract. Video understanding requires models to continuously track and update world state during playback. Although existing benchmarks have advanced video understanding evaluation across multiple dimensions, they provide limited visibility into how models maintain world state over time. We propose SVCBench, a Streaming Video Counting Benchmark that repositions counting as a minimal, controlled probe for diagnosing models' world-state maintenance capability. We decompose this capability into object counting and event counting, forming 8 fine-grained subcategories. Object counting covers tracking currently visible objects and cumulative unique identities, while event counting covers detecting instantaneous actions and tracking complete activity cycles. SVCBench contains 406 videos with frame-by-frame annotations of 10,071 event occurrences and object state changes, yielding 1,000 streaming QA pairs with 4,576 query points distributed along video timelines. By observing state maintenance trajectories through streaming multi-point queries, we design three complementary metrics to diagnose numerical precision, trajectory consistency, and temporal awareness. Evaluations of mainstream video-language models show that current models still exhibit significant deficiencies in spatial-temporal state maintenance, with especially poor performance on periodic event counting. SVCBench provides a diagnostic framework for measuring and improving state maintenance in video understanding systems. Our code and data are available at https://buaa-colalab.github.io/SVCBench.

1 Introduction

Consider the scenario in Figure 1. For the question "How many chairs are visible at this moment?", the answer changes as the camera moves. For the question

† Corresponding author

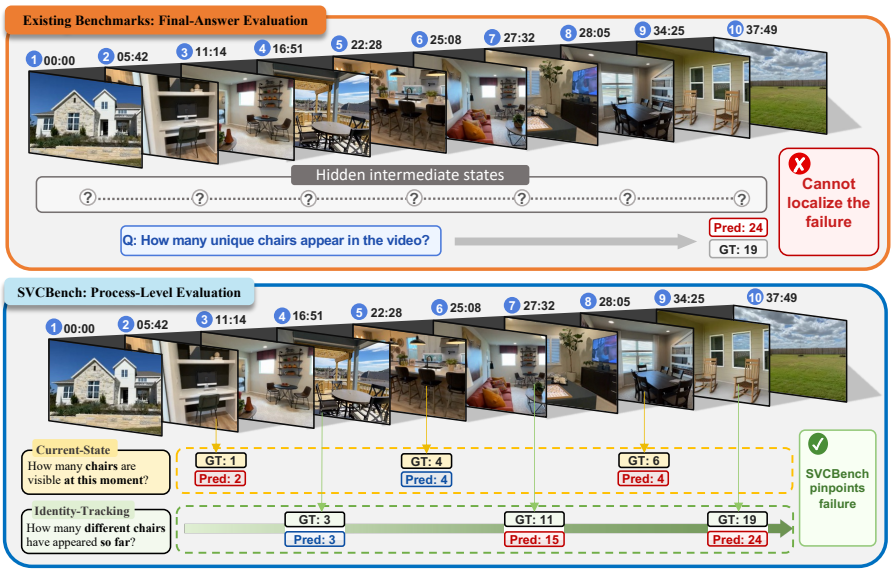


Fig. 1: Final-answer evaluation (top) only scores the last answer and cannot localize the error, whereas the process-level streaming queries in SVCBench (bottom) pinpoint when and where state maintenance fails.

"How many different chairs have appeared so far?", the answer should monotonically increase. These questions impose different requirements: the first requires tracking a current count that changes in real time; the second requires tracking cumulative unique identities. We refer to these internally tracked quantities, such as current counts, cumulative identities, and event occurrences, as **world state**, and the ability to continuously update them as **spatial-temporal state maintenance**. This ability is also closely related to embodied agents, which must continuously maintain and update task-relevant world state while perceiving and acting in dynamic environments. Throughout this paper, we use "state maintenance" as shorthand for spatial-temporal state maintenance. However, when models keep counts unchanged after objects leave, or produce decreasing counts in identity-tracking tasks, such contradictions expose defects in their state maintenance mechanisms.

Current video understanding benchmarks have made significant progress in standardization and capability coverage [8,12,16,23,25,27,30,35], paralleling the rapid advances of vision-language models more broadly [7,17,18]. However, we focus on an orthogonal dimension: **can existing evaluations observe how models maintain world state throughout video playback?** Most counting tasks present single queries where models provide a number after the video ends. Streaming benchmarks [14,16,29] introduce queries at multiple moments, but design choices—such as conflating tasks with different tracking requirements

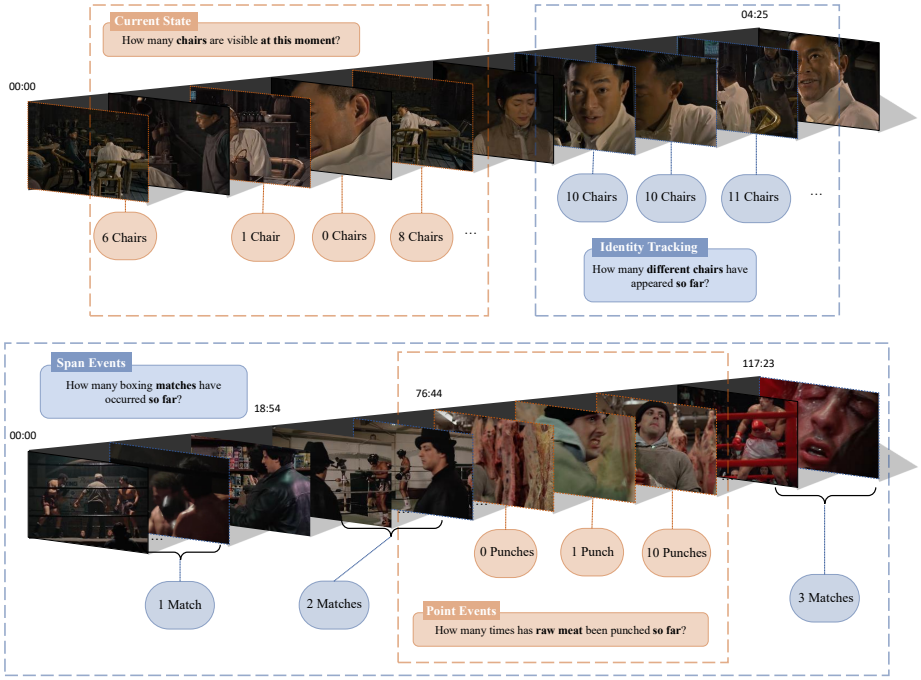


Fig. 2: Streaming counting as a probe for spatial-temporal state maintenance. Top: Object counting tracks the current count at each moment (current-state) and cumulative unique identities across time. Bottom: Event counting detects instantaneous actions and span events. Models are queried at multiple timepoints during video playback.

and using short temporal query windows—often prevent them from effectively evaluating models' spatial-temporal state maintenance. To fill this gap, we propose SVCBench, a **S**treaming **V**ideo **C**ounting **Bench**mark for long videos. Our core design principle is to **reposition counting as a minimal probe for diagnosing how models maintain internal representations over time**. Specifically, counting questions must depend on well-defined quantities (such as the current object count in a scene, or the number of distinct individuals seen so far) and are difficult to bypass through vague semantic recognition or language priors. Numerical answers provide deterministic, option-interference-free supervision while avoiding the judgment dilemma of open-ended QA. We position counting as a controlled diagnostic entry point rather than a complete characterization of state maintenance: it yields objective, verifiable ground truth (e.g., "14 chairs"), whereas many other states (e.g., "has the person finished cooking?") involve subjective boundaries or open-ended answers that are difficult to score reliably. More critically, by asking streaming questions at multiple timepoints during video playback, we can observe how models' predictions evolve rather than only checking isolated answers: if a model's prediction decreases from 5

to 3 in an identity tracking task, or fails to respond promptly to object entry/exit in a current-state task, such inconsistencies directly expose defects in its tracking mechanism. Based on this principle, we construct a systematic capability taxonomy that divides object counting into tracking currently visible objects and tracking cumulative unique identities, and event counting into detecting instantaneous actions and tracking complete activity cycles (Figure 2); these dimensions are further subdivided into eight fine-grained subcategories.

SVCBench contains 406 videos covering indoor navigation, first-person activities, sports videos, and other diverse scenes. We perform frame-by-frame annotation of 10,071 event occurrences and object state changes, and design 4,576 streaming query points across 1,000 QA pairs to observe how models' predictions evolve over time. We propose three complementary evaluation metrics to measure numerical precision, trajectory consistency, and temporal awareness. Systematic evaluation on mainstream video understanding models reveals significant deficiencies in current models' spatial-temporal state maintenance, providing clear directions for future model design.

2 Related Work

Long-Video Understanding Benchmarks. Video-MME [8] achieves a highly standardized comprehensive evaluation framework through multiple-choice questions, establishing full-spectrum coverage across multimodal tasks. LVBench [25] pushes evaluation of long-term memory and understanding capabilities with extremely long videos (up to several hours). MLVU [35] systematically evaluates long video understanding through duration spans from 3 minutes to 2 hours with multi-task settings. LongVideoBench [27] focuses on referential reasoning in long contexts, where the core difficulty lies in retrieving referenced segments and completing reasoning. These benchmarks have advanced long video semantic understanding, but their evaluation signals are mostly task-level one-time answers, making it difficult to observe how models maintain world state along the timeline.

Streaming and Online Video Evaluation. OVO-Bench [16], as the most systematic online evaluation framework, proposes Backward Tracing, Real-Time Perception, and Forward Active Responding modes, evaluating models' dynamic reasoning and temporal awareness through precise timestamped queries at each timepoint. StreamingBench [14] focuses on understanding performance under real video stream input, covering real-time visual understanding, full-source information fusion, and contextual understanding subtasks. RTV-Bench [29] includes hundreds of hours of videos and thousands of QA samples, fine-grained testing models' responses to temporal information through extensive timestamp data. These benchmarks have advanced online video understanding evaluation, but their queries are often distributed within relatively short time windows, making it difficult to impose long-term cumulative state maintenance pressure on models, and existing online counting tasks often do not explicitly distinguish

different tracking requirements for instantaneous actions versus extended activities.

Counting and Temporal Reasoning Evaluation. TOMATO [23] constructs multi-task evaluation including action count, measuring temporal dependencies with metrics like multi-frame gain and sequence sensitivity. VSI-Bench [30] integrates counting into spatial intelligence evaluation, examining spatial relationships and object quantities under continuous observation of indoor scenes. CountLLM [32] focuses on repetitive action counting, emphasizing the role of periodic prompts and language supervision for counting generalization. AV-Reasoner [15] supports interpretable counting evaluation through clue annotation. These works have advanced counting and temporal reasoning evaluation from different perspectives, but most benchmarks emphasize temporal relationships within segments or specific counting subtasks, rarely making long-term cumulative state maintenance pressure and continuous updating of world state core design goals.

Complementary to existing work, SVCBench reformalizes counting as a minimal probe of world state, observing models' state maintenance trajectories through streaming multi-point queries, and explicitly aligning different types of state maintenance requirements to diagnosable sub-capabilities through a systematic capability decomposition framework.

3 The SVCBench Benchmark

3.1 Taxonomy: Decomposing State Maintenance via Counting

We decompose spatial-temporal state maintenance capability into two orthogonal dimensions: object counting and event counting. Object counting focuses on tracking entities in the scene, while event counting focuses on identifying and tracking actions or activities in videos. Within each dimension, we subdivide tasks according to which quantities must be tracked and how those quantities update over time, forming the taxonomy shown in Figure 3.

Object Counting. Object counting tracks entities in the scene. Based on query semantics, we divide it into current-state object counting (O1) and identity-tracking object counting (O2).

Current state (O1). O1 requires tracking the current count as it changes in real time. O1-Snap queries the number of objects visible in the scene at a given moment, so the model must update the count as objects enter or leave the view. O1-Delta queries the change relative to the start of a time window, which requires the model to retain the earlier count and compute the difference.

Cumulative identities (O2). O2 requires tracking cumulative unique identities across time. O2-Unique queries the total number of distinct individuals that have appeared across the timeline, so the model must judge whether each newly seen individual is new or a repeat. O2-Gain queries how many new individuals appear within a time window, combining identity tracking with a windowed constraint.

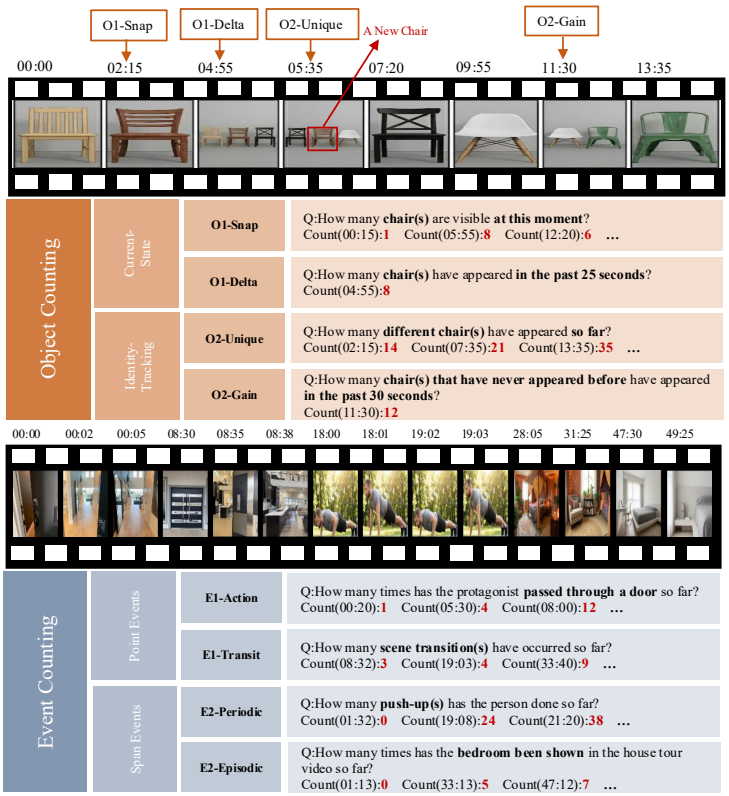


Fig. 3: Taxonomy of spatial-temporal state maintenance. We decompose counting into 8 subcategories across object and event dimensions. Each subcategory requires tracking different types of information: O1 tracks currently visible quantities, O2 tracks cumulative unique identities, E1 counts discrete instantaneous occurrences, and E2 counts complete activity cycles.

Event Counting. Event counting identifies and tracks actions or activities over time. Based on temporal structure, we divide it into point event counting (E1) for instantaneous occurrences and span event counting (E2) for activities with temporal extent.

Point events (E1). E1 focuses on instantaneous atomic actions that can be approximated as a single point on the timeline. E1-Action queries the cumulative number of times a given atomic action occurs, where each occurrence is brief and acts as a discrete pulse. E1-Transit queries the number of scene transitions, where the model must detect the moment the scene switches from one stable configuration to another while maintaining the running total.

Span events (E2). E2 focuses on activities with clear start and end moments that occupy an interval on the timeline. E2-Periodic queries the number of complete, repetitive action cycles, which requires identifying the start and end boundaries

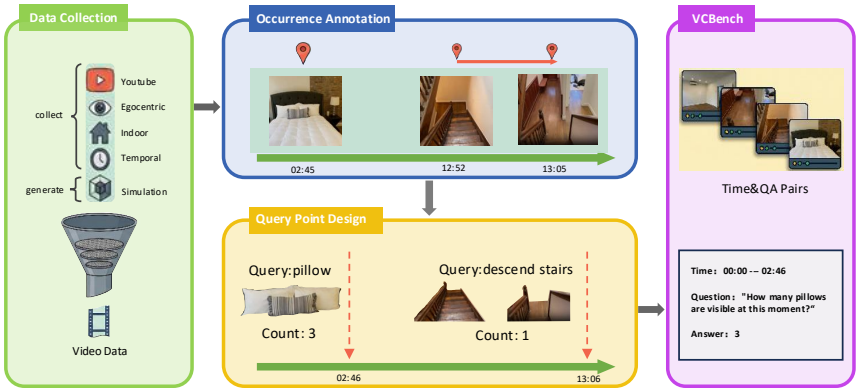


Fig. 4: SVCBench dataset construction pipeline. It shows the complete process from video source selection and fine-grained annotation generation to streaming query-point design. We collect videos from multiple sources including web platforms, existing computer vision datasets, and self-generated simulations, then perform manual annotations and design streaming query points.

of each cycle. E2-Episode queries the number of semantically complete activity segments, where the difficulty lies in boundary determination that depends on high-level semantic understanding.

O1 categories test models' ability to track current counts in real time, with prediction trajectories that can rise or fall; O2 categories test cross-temporal identity deduplication capability, with prediction trajectories that should be monotonically non-decreasing. E1 categories test precise moment detection capability, while E2 categories further require tracking complete event lifecycles. By observing models' prediction trajectories at multiple query points along the timeline, we can diagnose specific weaknesses in state initialization, incremental updates, and cross-temporal queries.

3.2 Dataset Construction

Figure 4 outlines the construction pipeline, from video sourcing through frame-by-frame annotation to streaming query-point design.

Video Sources and Special Processing. We collect videos from multiple sources: web platforms including YouTube (108 videos), existing computer vision datasets including ARKitScenes [3], ScanNet [6], ScanNet++ [33], Ego4D [10], RoomTour3D [11], TOMATO [23], CODa [13], and OmniWorld [36], as well as self-generated physics-simulation videos. For the E2-Periodic category, we carefully selected 56 periodic action videos from TOMATO with seamless loop points and extended them through loop concatenation to generate 2–3 minute videos, addressing the scarcity of long-duration periodic video data.

Annotation Protocol. The annotation process contains two core components. First, we perform frame-by-frame annotation of event occurrences and object

Table 1: Dataset statistics by subcategory. Annotated events are independent occurrences. E2 events include start and end timestamps. Avg. Density denotes the average number of queried targets per QA; "-" marks entries that are undefined for state-querying categories (O1-Snap/O1-Delta/O2-Gain), whose answers correspond to an instantaneous count or a window difference rather than enumerated events.

Category	Questions	Videos	Annotated Events	Avg. Density
Object Counting	**546**	**196**	**2,483**	**4.5**
O1-Snap	81	78	-	-
O1-Delta	98	75	-	-
O2-Unique	289	171	2,483	8.6
O2-Gain	78	64	-	-
Event Counting	**454**	**281**	**7,588**	**16.7**
E1-Action	203	146	1,893	9.3
E1-Transit	48	29	682	14.2
E2-Episode	147	100	1,033	7.0
E2-Periodic	56	56	3,980	71.1

state changes. Specifically, we annotate the timestamp of each occurrence for E1, the start and end moments for E2, and the first appearance moment of each individual for O2. Second, we design query points along the timeline, selecting moments with stable frames to avoid visual ambiguity. The correct answer for each query point is calculated based on annotated moments before that moment. O1-Delta and O2-Gain each set only 1 query point per question, as they focus on changes within specific time windows.

Dataset Statistics. SVCBench contains 406 videos, 1,000 counting questions, 10,071 precisely annotated event occurrence moments and object state change moments, and 4,576 query points. Table 1 shows detailed statistics for each subcategory. Figure 5 further illustrates the distribution of video sources, the diversity of counting-target semantics, and the distribution of video durations.

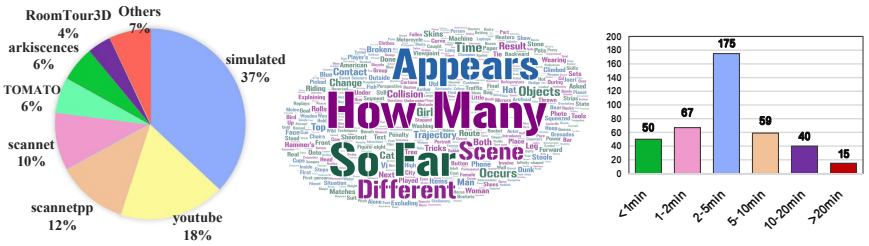
3.3 Evaluation Metrics

SVCBench's streaming multi-point queries produce prediction trajectories rather than isolated answers. We design three complementary metrics to measure numerical precision, trajectory consistency, and temporal awareness.

GPA (Gaussian Precision Accuracy) measures the match between predictions and correct answers. For a question q with n query points, prediction sequence $P = [p_1, \ldots, p_n]$, and correct answer sequence $G = [g_1, \ldots, g_n]$:

$$\text{GPA}_q = \frac{1}{n} \sum_{i=1}^{n} \exp\left(-\frac{(p_i - g_i)^2}{2\sigma_i^2} \right), \quad \sigma_i = 0.05 \cdot \max(g_i, 1) \tag{1}$$

By setting σ as 5% of the correct answer, the Gaussian kernel assigns near-zero scores to predictions with more than 15% relative deviation (3σ), effectively penalizing significant outliers. GPA $\in [0, 1]$ applies to all 8 subcategories.



(a) Video Sources (b) Query Semantics (c) Video Durations

Fig. 5: SVCBench dataset analysis. (a) Pie chart of video source distribution, showing scene diversity across data sources. (b) Word cloud of query target semantics, showing the diversity of counted objects and events. (c) Histogram of video duration distribution.

MoC (Monotonicity Consistency) verifies whether prediction sequences maintain the monotonic non-decreasing property for cumulative tasks. For a question q with $n \geq 2$ query points, define the first violation position $v = \min\{i : p_{i+1} < p_i\}$ (if none exists, $v = n$):

$$\mathrm{MoC}_q = \frac{v-1}{n-1} \tag{2}$$

$\mathrm{MoC} \in [0, 1]$, where 1 indicates a completely monotonic non-decreasing trajectory.

UDA (Update Detection Accuracy) measures whether models update their predictions at the moments when the world state actually changes. Let $S = \{i : g_i \neq g_{i-1}\}$ be the set of adjacent query points across which the ground-truth count changes. UDA is the fraction of these state-change steps at which the model also updates its prediction:

$$\mathrm{UDA}_q = \frac{1}{|S|} \sum_{i \in S} \mathbb{1}[p_i \neq p_{i-1}] \tag{3}$$

$\mathrm{UDA} \in [0, 1]$ (undefined when $|S| = 0$), where 1 indicates the model responds at every state-change moment. MoC checks whether the trajectory regresses, while UDA checks whether the model updates exactly when the ground truth changes.

The three metrics form a complementary diagnostic framework: GPA captures numerical precision, MoC captures trajectory consistency, and UDA captures update timing. Their combination distinguishes failure modes. For instance, high GPA and MoC but low UDA indicates numerically accurate, monotonic predictions whose updates lag the true state changes, whereas low GPA together with low MoC indicates large numerical deviation and an unstable trajectory.

4 Experiments

4.1 Experimental Setup

Evaluated Models. We evaluate three types of systems to comprehensively compare spatial-temporal state maintenance capabilities under different architectural paradigms. **Offline multimodal models** include proprietary models (Gemini-3-Flash [9], Doubao-Seed-1.8 [4], Kimi-K2.5 [24], GPT-5.4 [19]) and open-source models (Qwen3-VL-8B/30B [2], Qwen2.5-VL-7B [21], InternVL-3.5-8B [26], Molmo2-8B [5], and the Mixture-of-Experts Qwen3.5-35B-A3B [22]). These models independently process videos truncated to each query point. **Online video understanding models** include StreamingVLM [28] (with a Qwen2.5-VL-7B backbone), Dispider [20], LiveStar [31], and Flash-VStream-7B [34], which process video frames in a streaming manner, continuously updating internal state and responding to queries at any moment without reprocessing historical frames. We also introduce **GPT-4-Turbo** [1] as a blind language model baseline, receiving only text questions without video frames, to quantify the actual contribution of visual information. **Human agents** complete evaluation under the same streaming query setting as models, watching videos and pausing to answer at each query point without rewatching played segments, providing capability upper bounds for model evaluation.

Implementation Details. Offline models are evaluated using the protocol proposed by OVO-Bench [16]: for each query point, videos are truncated to that moment as input based on the model's video input limitations. For proprietary models with a fixed frame budget (Doubao-Seed-1.8, Kimi-K2.5, and GPT-5.4), we uniformly sample 64 frames from the truncated video at each query point, whereas API models that accept native frame rates (Gemini-3-Flash) are queried at 1 fps up to their context limit. The query format is: "Based on the video content up to this moment, [question] Please answer with a single number." Online models process video streams according to their native inference flow, injecting query questions upon reaching query moments and directly generating answers based on maintained internal state. All model outputs undergo unified post-processing to extract numbers, supporting both Arabic numerals and English number words; extraction failures are marked as invalid answers and scored as incorrect, contributing zero to GPA, so that a model is not rewarded for failing to produce a parseable count.

4.2 Main Results

Tables 2 and 3 show complete results for all evaluated models on SVCBench.

Significant Human-Machine Gap. Human agents achieve GPA between 93–100 across all subcategories, while the best models (Gemini-3-Flash [9] and Doubao-Seed-1.8 [4]) achieve overall GPA of only around 37, indicating spatial-temporal state maintenance is a broad bottleneck for current video understanding models. Notably, humans achieve GPA of 93.2 on E2-Periodic, whereas the

Table 2: Overall Performance and Object Counting Results on SVCBench. GPA: Gaussian Precision Accuracy, MoC: Monotonicity Consistency, UDA: Update Detection Accuracy (scaled to 0–100). Best in **bold**, second underlined.

Model	Fr.	Overall			O1-Snap		O1-Delta	O2-Unique			O2-Gain
		GPA	MoC	UDA	GPA	UDA	GPA	GPA	MoC	UDA	GPA
Human Performance											
Human	-	96.1	100.0	99.3	96.7	100.0	100.0	94.5	100.0	98.8	100.0
Blind LLMs											
GPT-4-Turbo [1]	-	18.7	95.7	4.3	15.7	0.6	19.4	15.8	96.4	3.6	50.0
Proprietary Multimodal Models											
Gemini-3-Flash [9]	1fps	**37.0**	73.7	73.8	**44.3**	64.4	41.0	**36.5**	**77.8**	**79.8**	55.3
Doubao-Seed-1.8 [4]	64	36.2	**77.2**	**76.8**	43.4	**70.4**	**53.3**	32.5	75.9	**79.8**	**69.2**
Kimi-K2.5 [24]	64	26.4	66.8	73.4	28.2	63.6	18.4	29.9	62.9	78.2	24.4
GPT-5.4 [19]	64	29.1	71.8	59.3	26.6	45.5	26.5	30.9	72.2	57.1	35.9
Open-Source Multimodal Models - Offline											
Qwen3-VL-8B [2]	64	**31.0**	84.3	55.1	21.1	**43.8**	**35.7**	34.2	**87.1**	58.0	**55.1**
Qwen3-VL-30B [2]	64	27.0	**84.6**	56.4	19.1	43.2	23.5	33.1	84.4	63.9	26.9
Qwen2.5-VL-7B [21]	64	19.1	68.2	40.8	**29.7**	38.3	20.4	23.8	67.1	53.2	46.2
InternVL-3.5-8B [26]	64	24.2	81.6	55.6	20.3	37.4	6.1	30.5	82.2	57.8	37.2
Molmo2-8B [5]	64	8.5	66.2	34.6	26.5	42.6	1.0	13.0	53.4	46.8	3.8
Qwen3.5-35B-A3B [22]	64	29.3	82.4	**59.0**	20.9	41.6	28.8	**37.8**	85.6	**70.8**	53.8
Open-Source Multimodal Models - Online											
StreamingVLM [28]	1fps	19.1	68.1	**50.3**	**36.7**	**60.3**	6.1	**26.0**	65.6	55.4	14.1
Dispider [20]	1fps	7.7	34.8	15.5	11.4	8.2	16.3	3.3	34.8	16.8	3.8
LiveStar [31]	1fps	19.9	**86.4**	36.8	24.1	16.0	13.3	20.9	**81.8**	**57.3**	26.9
Flash-VStream-7B [34]	1fps	**23.4**	78.5	34.7	14.8	36.6	**45.9**	21.9	80.8	37.9	**56.4**

best model on this subcategory reaches only 11.1 and almost all others fall below 4, highlighting the difficulty of periodic event counting.

Overall Advantage of Proprietary Models. Proprietary models generally outperform open-source models in GPA (37 vs 27–31 for best open-source models), but open-source models show unexpected advantages in MoC (84+ vs 67–77), indicating open-source models tend to output smooth monotonic sequences but lack numerical precision. This may reflect different training strategies: proprietary models may be more aggressive in predicting large changes, but also more prone to violating monotonicity constraints. Greater scale or general capability does not by itself yield better state maintenance. GPT-5.4 [19] reaches only 29.1 overall GPA, below Gemini-3-Flash and Doubao-Seed-1.8, and the Mixture-of-Experts Qwen3.5-35B-A3B [22] leads open-source models on update timing (overall UDA 59.0) but still lags in numerical precision (GPA 29.3).

High MoC of Blind Baseline. GPT-4-Turbo [1] achieves MoC of 95.7 without any visual input, indicating pure language models understand the language prior that cumulative counts should increase. However, its GPA is only 18.7 and its UDA is only 4.3, indicating visual information is crucial for accurate state maintenance and timing perception.

Trade-offs Between Online and Offline Models. At comparable parameter scale, online models do not uniformly underperform offline ones in overall GPA. Flash-VStream-7B (7B) reaches 23.4, the strongest in overall GPA among the online models we evaluate and comparable to or better than 7–8B offline

Table 3: Event Counting Results on SVCBench. GPA: Gaussian Precision Accuracy, MoC: Monotonicity Consistency, UDA: Update Detection Accuracy (scaled to 0–100). Best in **bold**, second underlined.

Model	E1-Action			E1-Transit			E2-Episode			E2-Periodic		
	GPA	MoC	UDA	GPA	MoC	UDA	GPA	MoC	UDA	GPA	MoC	UDA
Human Performance												
Human	94.9	100.0	99.3	98.3	100.0	100.0	97.0	100.0	99.3	93.2	100.0	100.0
Blind LLMs												
GPT-4-Turbo [1]	13.1	96.5	4.2	23.6	95.8	4.2	21.9	93.7	7.1	0.0	94.2	5.4
Proprietary Multimodal Models												
Gemini-3-Flash [9]	**28.5**	66.0	61.8	**49.8**	83.7	**87.3**	**41.7**	79.5	78.0	**3.9**	**56.2**	**88.8**
Doubao-Seed-1.8 [4]	24.0	**76.9**	**70.7**	38.2	**96.4**	82.1	40.5	**84.4**	**79.3**	0.8	50.4	81.7
Kimi-K2.5 [24]	21.9	66.0	70.3	33.5	74.8	83.9	38.7	78.7	74.3	0.4	52.2	63.4
GPT-5.4 [19]	24.2	71.5	58.5	42.0	85.9	82.1	37.5	79.1	70.6	3.2	40.2	45.1
Open-Source Multimodal Models - Offline												
Qwen3-VL-8B [2]	**22.5**	84.9	**54.5**	**36.9**	95.3	70.8	**35.6**	82.1	65.4	0.0	68.8	29.0
Qwen3-VL-30B [2]	**22.5**	**88.1**	46.4	33.6	92.7	75.9	35.3	**89.0**	64.5	**2.5**	68.8	**37.5**
Qwen2.5-VL-7B [21]	9.5	66.7	33.3	13.3	90.6	23.4	11.0	66.5	37.9	0.0	65.2	32.1
InternVL-3.5-8B [26]	**22.5**	80.2	52.1	20.0	86.8	77.1	31.8	81.3	65.3	1.0	66.5	30.4
Molmo2-8B [5]	3.5	71.7	21.7	2.7	**95.5**	7.6	9.4	73.7	35.3	0.4	66.1	30.4
Qwen3.5-35B-A3B [22]	19.0	78.1	47.6	27.7	87.2	**80.6**	29.3	83.9	**67.1**	0.8	**73.1**	26.9
Open-Source Multimodal Models - Online												
StreamingVLM [28]	15.8	73.4	**40.8**	13.8	58.5	**61.3**	20.7	63.3	**54.9**	0.4	82.6	**24.1**
Dispider [20]	11.8	79.8	17.0	8.9	67.5	15.8	7.6	84.0	14.7	0.0	**96.4**	4.9
LiveStar [31]	16.1	**89.5**	23.5	**22.8**	**91.7**	31.9	**23.7**	**87.1**	38.8	**11.1**	93.2	8.9
Flash-VStream-7B [34]	**19.0**	75.5	31.0	11.9	84.2	41.8	17.2	77.0	35.7	0.0	76.4	19.9

models such as Qwen2.5-VL-7B (19.1). The online disadvantage instead emerges over time: when an online model is compared against its own offline backbone under identical query sequences, accuracy degrades faster along the trajectory (see the supplementary material), likely because they must compress historical information into fixed-size state representations. However, online models show higher MoC in some subcategories, indicating continuous internal state maintenance makes prediction trajectories smoother. Online models do not need to reprocess historical frames at each query point, which theoretically gives them computational advantages, but the cost is that state compression may lead to information loss.

4.3 Controlled Experiments on State Maintenance Failures

To deeply understand models' failure mechanisms in spatial-temporal state maintenance, we design two controlled experiments targeting cycle boundary detection in event counting and identity persistence tracking in object counting.

Visual-Temporal Grounding via Explicit Count Annotation E2-Periodic (counting complete repetitive action cycles) presents the most severe failure mode we observe: almost all evaluated models achieve GPA below 4 (the strongest offline model, Gemini-3-Flash [9], reaches only 3.9; even the best model overall, the online LiveStar [31], attains just 11.1), while human agents

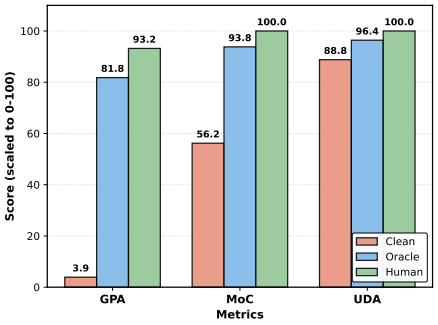


Fig. 6: Impact of explicit count annotation on E2-Periodic performance (scaled to 0–100).

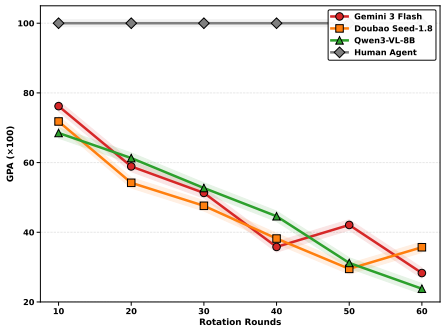


Fig. 7: Identity persistence degradation in rotating camera views (GPA scaled to 0–100).

achieve 93.2. Despite GPA approaching 0, models' MoC remains relatively high, indicating models are not randomly guessing but tend to output monotonically increasing sequences. This suggests models understand the language prior that cumulative counts should increase but largely fail to identify cycle boundaries from visual input.

To isolate visual perception capability from reasoning capability, we design an oracle experiment in which we evaluate Gemini-3-Flash [9] on 56 E2-Periodic questions from the TOMATO dataset [23], comparing clean videos with videos overlaid with explicit cycle count annotations in the upper-left corner (e.g., "Current count: 5"). This annotation directly tells the model the number of completed cycles, effectively bypassing the visual-temporal alignment bottleneck and directly testing the model's numerical reasoning capability.

As Figure 6 shows, explicit annotation improves GPA from 3.9 to 81.8 (21×), approaching human level (93.2). MoC improves by 37.6 points, with monotonicity violations dropping from 43.8% to only 6.2%. This large improvement suggests that models possess sufficient reasoning capability to maintain cumulative counts, but largely lack the ability to detect cycle boundaries from pure visual input. In other words, the bottleneck is not understanding that counts should increase, but recognizing when they should be incremented. This indicates current video-language models lack robust visual-temporal grounding mechanisms and struggle to map repetitive visual motion patterns to discrete cycle boundaries.

Natural long-period videos. To verify that this failure is not an artifact of loop concatenation, we additionally collect 50 natural long-period videos from YouTube (sports activities, 2–5 min, with natural cycle variation). Model performance on these natural videos remains low and consistent with the loop-concatenated set (e.g., Gemini-3-Flash 3.9→2.9, Doubao-Seed-1.8 0.8→0.5 GPA), while humans stay high (93.2→91.5), suggesting the periodic-counting bottleneck is not solely an artifact of loop concatenation.

Identity Persistence Degradation in Rotating Camera Views To test models' persistence maintenance capability in object identity tracking, we construct a controlled scenario in which a camera rotates 360 degrees in place indoors, with only 1 bench remaining stationary. As the camera rotates, the bench continuously enters and leaves the field of view, but these are all repeated appearances of the same bench. We query "How many different benches have appeared so far?" The correct answer should remain 1 throughout.

We let the camera rotate continuously for 60 cycles and calculate the average GPA over all query points within each 10-cycle interval to observe the degradation trend of models' identity persistence maintenance capability over time. Figure 7 shows the performance of three mainstream models.

In the first 10 cycles, the three models achieve GPA of 76.2 (Gemini-3-Flash [9]), 71.8 (Doubao-Seed-1.8 [4]), and 68.5 (Qwen3-VL-8B [2]). As rotation cycles increase, all three models' GPA shows a downward trend with notable fluctuations. By 60 cycles, the three models' GPA drops to 28.3, 35.7, and 23.8 respectively, approaching random guessing levels, while human agents maintain perfect performance of 100.

This degradation curve indicates that models struggle to maintain a robust cross-temporal identity-persistence representation of the set of seen objects. As the number of object reappearances increases, models' internal state representation gradually degrades, tending to misidentify each reappearing bench as a new object. This experiment echoes the count-annotation experiment above: the former exposes failure in event boundary detection, the latter exposes failure in object identity maintenance, both suggesting current video-language models have limited representations for continuously updatable world state.

5 Conclusion

We propose SVCBench, a streaming counting benchmark for systematically measuring the spatial-temporal state maintenance capability of video understanding models. By reformalizing counting as a minimal probe of world state, we establish a systematic taxonomy that decomposes state maintenance capability into fine-grained subcategories across object and event dimensions. Our streaming multi-point query design enables evaluation to observe prediction trajectories rather than isolated predictions, thereby revealing failure modes that final-answer evaluation is unable to surface.

Systematic evaluation on mainstream models exposes clear limitations. The pronounced failure on periodic event counting suggests current architectures lack robust mechanisms for tracking temporal structure from visual input. Our oracle experiment provides mechanistic insight: the 21× improvement brought by explicit count annotation suggests models possess reasoning capability but lack visual-temporal grounding. Similarly, the rotating camera experiment indicates models struggle to maintain identity persistence, often misidentifying repeatedly appearing objects as new entities. Together, these findings point to limitations

in current architectures: models have learned language priors but largely fail to ground them in world state maintenance.

SVCBench provides a diagnostic framework for advancing video understanding. Our taxonomy and metric system provide actionable improvement targets. The benchmark's streaming evaluation protocol and fine-grained subcategories enable researchers to isolate specific failure modes and measure progress in state maintenance capabilities. We hope SVCBench will support the development of video models with genuine temporal reasoning capabilities, advancing from pattern matching toward robust world state tracking.

Acknowledgement. This research is supported in part by the National Natural Science Foundation of China (No. 62461160308, U23B2010, 62576024), the Beijing Natural Science Foundation (No. L231011), the Fundamental Research Funds for the Central Universities (No. 501RCQD2025141003), and BeiHang GanWei Project (No. 502GWXM2024141001). We also acknowledge the support of the Beijing Nova Program.

References

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)
2. Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., et al.: Qwen3-vl technical report. arXiv preprint arXiv:2511.21631 (2025)
3. Baruch, G., Chen, Z., Dehghan, A., Dimry, T., Feigin, Y., Fu, P., Gebauer, T., Joffe, B., Kurz, D., Schwartz, A., et al.: Arkitscenes: A diverse real-world dataset for 3d indoor scene understanding using mobile rgb-d data. In: Advances in Neural Information Processing Systems Datasets and Benchmarks Track (2021)
4. ByteDance: Doubao seed 1.8. `https://www.volcengine.com/product/doubao` (Dec 2025), accessed: 2026-03
5. Clark, C., Zhang, J., Ma, Z., Park, J.S., Salehi, M., Tripathi, R., Lee, S., Ren, Z., Kim, C.D., Yang, Y., et al.: Molmo2: Open weights and data for vision-language models with video understanding and grounding. arXiv preprint arXiv:2601.10611 (2026)
6. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5828–5839 (2017)
7. Dong, H., Niu, J., Wang, B., Zeng, W., Zhang, W., He, C.: MinerU-diffusion: Rethinking document OCR as inverse rendering via diffusion decoding. arXiv preprint arXiv:2603.22458 (2026)
8. Fu, C., Dai, Y., Luo, Y., Li, L., Ren, S., Zhang, R., Wang, Z., Zhou, C., Shen, Y., Zhang, M., et al.: Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 24108–24118 (2025)
9. Google: A new era of intelligence with gemini 3. `https://blog.google/products-and-platforms/products/gemini/gemini-3/` (2025), accessed: 2026-03-05

10. Grauman, K., Westbury, A., Byrne, E., Chavis, Z., Furnari, A., Girdhar, R., Hamburger, J., Jiang, H., Liu, M., Liu, X., et al.: Ego4d: Around the world in 3,000 hours of egocentric video. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 18995–19012 (2022)
11. Han, M., Ma, L., Zhumakhanova, K., Radionova, E., Zhang, J., Chang, X., Liang, X., Laptev, I.: Roomtour3d: Geometry-aware video-instruction tuning for embodied navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 27586–27596 (2025)
12. Han, S., Huang, W., Shi, H., Zhuo, L., Su, X., Zhang, S., Zhou, X., Qi, X., Liao, Y., Liu, S.: Videoespresso: A large-scale chain-of-thought dataset for fine-grained video reasoning via core frame selection. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 26181–26191 (2025)
13. Li, K., Chen, K., Wang, H., Hong, L., Ye, C., Han, J., Chen, Y., Zhang, W., Xu, C., Yeung, D.Y., et al.: Coda: A real-world road corner case dataset for object detection in autonomous driving. In: European Conference on Computer Vision. pp. 406–423. Springer (2022)
14. Lin, J., Fang, Z., Chen, C., Wan, Z., Luo, F., Li, P., Liu, Y., Sun, M.: Streamingbench: Assessing the gap for mllms to achieve streaming video understanding. arXiv preprint arXiv:2411.03628 (2024)
15. Lu, L., Chen, G., Wei, Z., Li, Z., Liu, Y., Lu, T.: AV-Reasoner: Improving and benchmarking clue-grounded audio-visual counting for MLLMs. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 33477–33487 (2026)
16. Niu, J., Li, Y., Miao, Z., Ge, C., Zhou, Y., He, Q., Dong, X., Duan, H., Ding, S., Qian, R., et al.: Ovo-bench: How far is your video-llms from real-world online video understanding? In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 18902–18913 (2025)
17. Niu, J., Liu, Z., Gu, Z., Wang, B., Ouyang, L., Zhao, Z., Chu, T., He, T., Wu, F., Zhang, Q., et al.: Mineru2. 5: A decoupled vision-language model for efficient high-resolution document parsing. In: Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (ACL 2026). pp. 13–42 (2026)
18. Niu, J., Zheng, Y., Miao, Z., Dong, H., Ge, C., Liang, H., Lu, M., Zeng, B., Zheng, Q., He, C., et al.: Native visual understanding: Resolving resolution dilemmas in vision-language models. In: Proceedings of the Computer Vision and Pattern Recognition Conference (2026)
19. OpenAI: Gpt-5.4 model. `https://developers.openai.com/api/docs/models/gpt-5.4` (2026), model ID: gpt-5.4; snapshot: gpt-5.4-2026-03-05. Accessed: 2026-03-25
20. Qian, R., Ding, S., Dong, X., Zhang, P., Zang, Y., Cao, Y., Lin, D., Wang, J.: Dispider: Enabling video llms with active real-time interaction via disentangled perception, decision, and reaction. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 24045–24055 (2025)
21. Qwen, :, Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Li, C., Liu, D., et al.: Qwen2.5 technical report. `https://arxiv.org/abs/2412.15115` (2025), accessed: 2026-03-25
22. Qwen Team: Qwen3.5. `https://qwen.ai/blog?id=qwen3.5` (2026), qwen3.5-35B-A3B (Mixture-of-Experts, 35B total / 3B active); released 2026-02-24. Accessed: 2026-03-25
23. Shangguan, Z., Li, C., Ding, Y., Zheng, Y., Zhao, Y., Fitzgerald, T., Cohan, A.: Tomato: Assessing visual temporal reasoning capabilities in multimodal foundation

models. In: International Conference on Learning Representations. vol. 2025, pp. 7593–7734 (2025)
24. Team, K., Bai, T., Bai, Y., Bao, Y., Cai, S., Cao, Y., Charles, Y., Che, H., Chen, C., Chen, G., et al.: Kimi k2. 5: Visual agentic intelligence. arXiv preprint arXiv:2602.02276 (2026)
25. Wang, W., He, Z., Hong, W., Cheng, Y., Zhang, X., Qi, J., Ding, M., Gu, X., Huang, S., Xu, B., et al.: Lvbench: An extreme long video understanding benchmark. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 22958–22967 (2025)
26. Wang, W., Gao, Z., Gu, L., Pu, H., Cui, L., Wei, X., Liu, Z., Jing, L., Ye, S., Shao, J., et al.: Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. arXiv preprint arXiv:2508.18265 (2025)
27. Wu, H., Li, D., Chen, B., Li, J.: Longvideobench: A benchmark for long-context interleaved video-language understanding. Advances in Neural Information Processing Systems **37**, 28828–28857 (2024)
28. Xu, R., Xiao, G., Chen, Y., He, L., Peng, K., Lu, Y., Han, S.: Streamingvlm: Real-time understanding for infinite video streams. arXiv preprint arXiv:2510.09608 (2025)
29. Xun, S., Tao, S., Li, J., Shi, Y., Lin, Z., Zhu, Z., Yan, Y., Li, H., Zhang, L., Wang, S., et al.: Rtv-bench: Benchmarking mllm continuous perception, understanding and reasoning through real-time video. Advances in Neural Information Processing Systems **38** (2026)
30. Yang, J., Yang, S., Gupta, A.W., Han, R., Fei-Fei, L., Xie, S.: Thinking in space: How multimodal large language models see, remember, and recall spaces. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 10632–10643 (2025)
31. Yang, Z., Zhang, K., Hu, Y., Wang, B., Qian, S., Wen, B., Yang, F., Gao, T., Dong, W., Xu, C.: Livestar: Live streaming assistant for real-world online video understanding. In: Advances in Neural Information Processing Systems (NeurIPS) (2025)
32. Yao, Z., Cheng, X., Huang, Z., Li, L.: Countllm: Towards generalizable repetitive action counting via large language model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19143–19153 (2025)
33. Yeshwanth, C., Liu, Y.C., Nießner, M., Dai, A.: Scannet++: A high-fidelity dataset of 3d indoor scenes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12–22 (2023)
34. Zhang, H., Wang, Y., Tang, Y., Liu, Y., Feng, J., Jin, X.: Flash-vstream: Efficient real-time understanding for long video streams. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2025)
35. Zhou, J., Shu, Y., Zhao, B., Wu, B., Liang, Z., Xiao, S., Qin, M., Yang, X., Xiong, Y., Zhang, B., et al.: Mlvu: Benchmarking multi-task long video understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13691–13701 (2025)
36. Zhou, Y., Wang, Y., Zhou, J., Chang, W., Guo, H., Li, Z., Ma, K., Li, X., Wang, Y., Zhu, H., et al.: Omniworld: A multi-domain and multi-modal dataset for 4d world modeling. arXiv preprint arXiv:2509.12201 (2025)